Internet Engineering

Sadegh Aliakbary

Advanced Material

Outline

- Information Retrieval
- Data Mining
- Data-warehouse & OLAP
- Big Data and NoSQL Databases

Information Retrieval and Text Search

Information Retrieval (IR)

- Process of retrieving **documents**
  - From a **collection**
  - In response to a **query** by a user
- Discipline that deals with the structure, analysis, organization, storage, searching, and retrieval of information
- Deals with **Unstructured Data**
- Example: Google

Query

» User’s information need:

» Expressed as a **free-form search request**

» Example:

Internet Engineering

“Internet Engineering”

“Internet * Engineering”

Java “Open Source” –apache

Types of Queries

- Keywords
- Phrases
- Boolean Operators
- Wildcards
- …

Types of Search Engines

- Web Search Engines

- Enterprise search systems
  - IR solutions for searching different entities in an enterprise’s intranet
  - Applications?

- Desktop search engines
  - Retrieve files, folders, and different kinds of entities stored on the computer

Accuracy of Search ???

» **Recall**

$$\text{recall} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{relevant documents}\}|}$$

» **Precision**

$$\text{precision} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|}$$

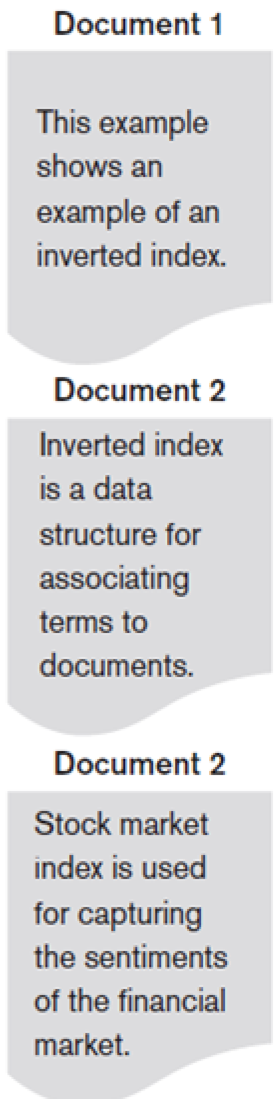
» **F-score**

- Single measure that combines precision and recall

$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

Inverted Indexing

- How to effic
- Vocabulary
  - Set of disti
- **Inverted in**
  - Data struct
    a list of all



ID	Term	Document: position
1.	example	1:2, 1:5
2.	inverted	1:8, 2:1
3.	index	1:9, 2:2, 3:3
4.	market	3:2, 3:13

Overview of IR Concepts

- Hyperlinks
- Crawler
- Vector Space Model
  - TF-IDF
- Ranking
  - Hubs and popular nodes
  - PageRank, Hits, …
- NLP tasks
  - Stop Words
  - Stemming

Search Result Steps

- Before Search
  - Query Processing
- After Search
  - Classification & Clustering
  - Query Expansion
  - Query Suggestion
    - Utilizing a Thesaurus: WordNet, ..

IR and Databsaes

- Support of text search in modern databases
  - Oracle Text
  - SQL Server Full-Text Search
- (Non-standard) SQL-extensions to support text search
  - Example:

select * from person where address like '%ولنجک%'

select * from person where geneder='male' AND CONTAINS (address, 'ولنجک')

- Other Technologies (not in a DBMS)
  - Lucene
  - Solr
  - Elastic Search

IR Summary

- Information Retrieval Concepts
  - Query
  - Inverted Index
  - Crawler
  - …
- IR and Databases
- IR Trends

Data Mining

Data Mining

- Data is the Value

- The value of many business are based on their gathered data
  - Banks, Social Networks, Online Services, ..

- Data Mining: Utilizing the value of the gathered data

Definitions of Data Mining

- The discovery of new information in terms of patterns or rules from vast amounts of data
- The process of finding **non-trivial** ? and **interesting** ? structure in data
- Based on Intelligent Algorithms
  - Artificial Intelligence
  - Machine Learning

Knowledge Discovery in Databases (KDD)

- Data mining is actually one step of a larger process known as KDD

- The KDD process model comprises six phases
  - Data selection
  - Data cleansing
  - Enrichment
    - Enhances the data with additional sources of information
  - Data transformation or encoding
  - Data mining
  - Reporting and displaying discovered knowledge

Types of Discovered Knowledge

- Association Rules
- Sequential Patterns
- Classification
  - Supervised Learning
- Clustering
  - Unsupervised Learning

Data Mining Methods

- Decision Tree
- K-Means
- KNN
- Neural Networks
- SVM
- …

Data Mining Applications

- Classification?
  - E.g., Customer Classification
- Clustering?
  - E.g., Search Results Clustering
- Association Rule Mining?
  - E.g., Product Suggestion

Data Mining and Databases

- Database is the base of the invaluable data

- Database and the Data Quality
  - DB Constraints
  - Clean Data : ready to be mined

- Data Mining Modules in DBMSs
  - E.g., Oracle Data Mining

Data Mining Summary

- Data Mining Concepts
- Methods
- Types of Knowledge
- Data Mining and Databases

Data Warehousing and OLAP

The Need to Datawarehousing

- There is a great need for tools that provide **decision makers** with information to make decisions quickly and reliably based on **historical data**.

- The above functionality is achieved by **Data Warehousing** and **Online analytical processing (OLAP)**

Purpose of Data Warehousing

- The **data warehouse** users need only **read access**
  - But, they need the access to be **fast over a large volume** of data
- The data in a data warehouse comes from **multiple databases**
- The analysis are recurrent and predictable
  - to be able to design specific software to meet the requirements
- **KPI**: Key Performance Indicators

Datawarehouse vs Database

- Datawarehouse:
- A **subject-oriented**, **integrated**, **nonvolatile**, **time-variant** collection of data in **support of management's decisions**

- An application-oriented, single, volatile, snapshot of data in support of a business operation

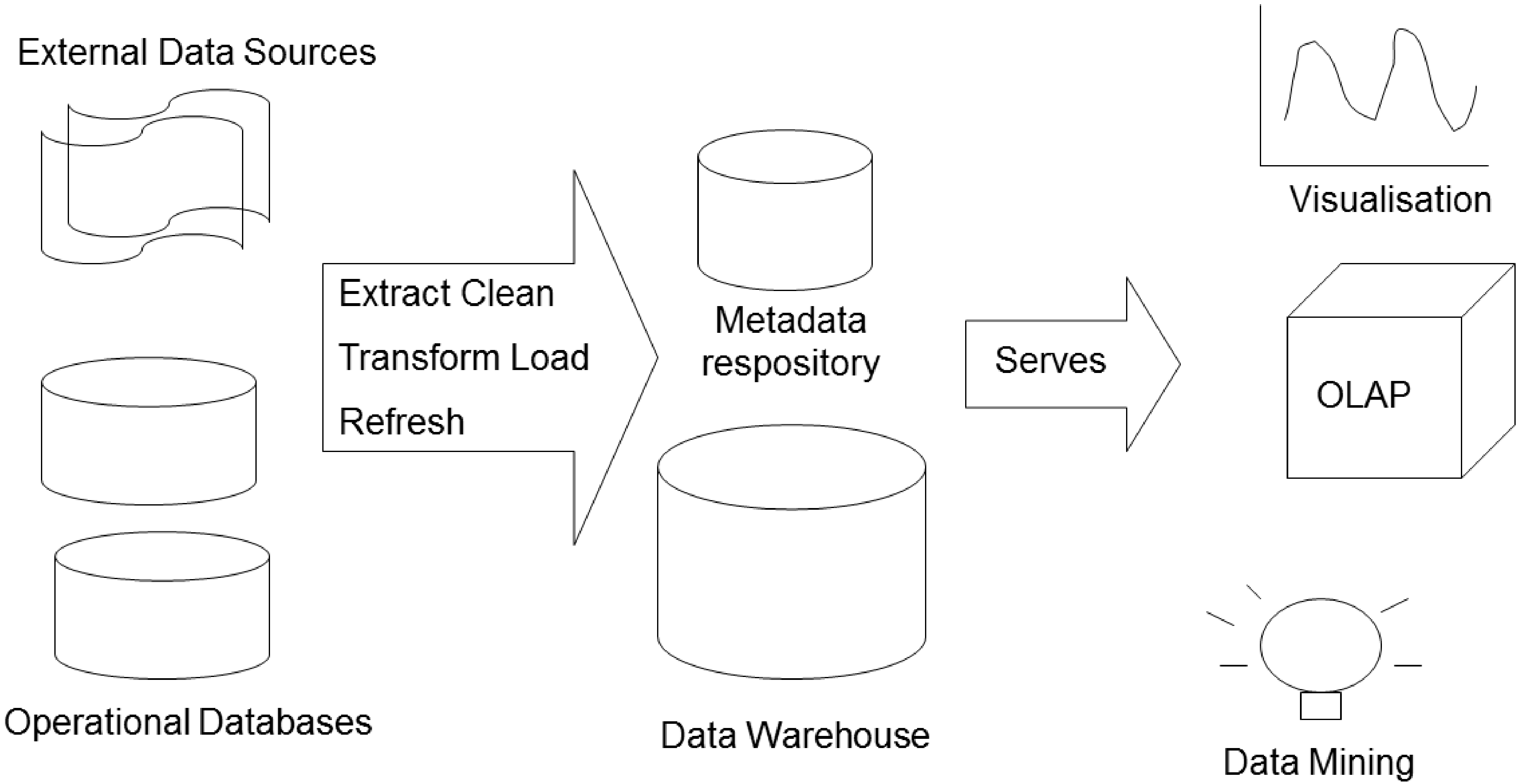
Applications of Datawarehouses

- **OLAP**
  - (Online Analytical Processing)
  - is a term used to describe the analysis of complex data from the data warehouse.
- **DSS**
  - (Decision Support Systems)
  - supports organization’s leading decision makers for making complex and important decisions
- **Data Mining**
  - is used for knowledge discovery, the process of searching data for unanticipated new knowledge.

OLTP vs OLAP

	OLTP	OLAP
Application	Operational: ERP, CRM, legacy apps, ...	Management Information System, Decision Support System
Typical users	Staff	Managers, Executives
Horizon	Weeks, Months	Years
Refresh	Immediate	Periodic
Data model	Entity-relationship	Multi-dimensional
Schema	Normalized	Star
Emphasis	Update	Retrieval

Conceptual Structure of Data Warehouse

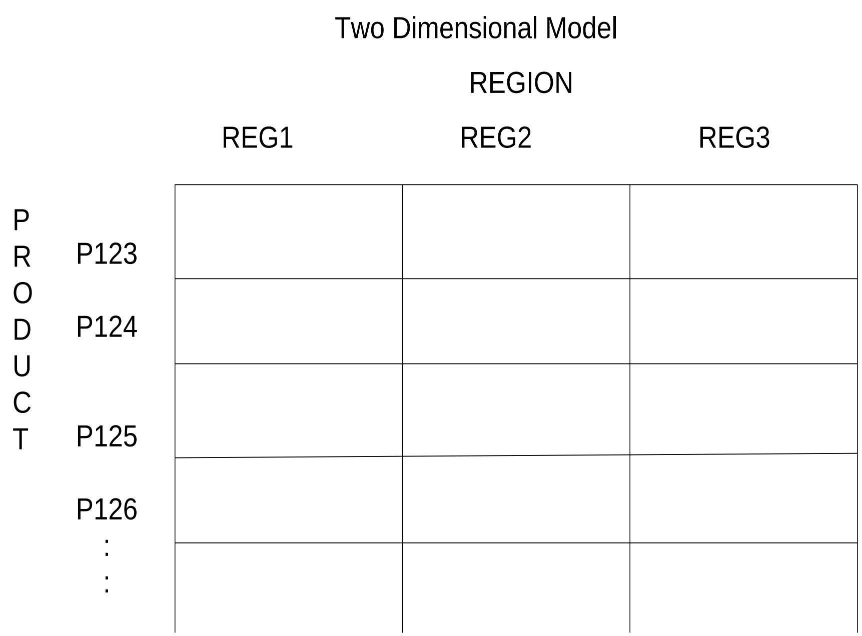


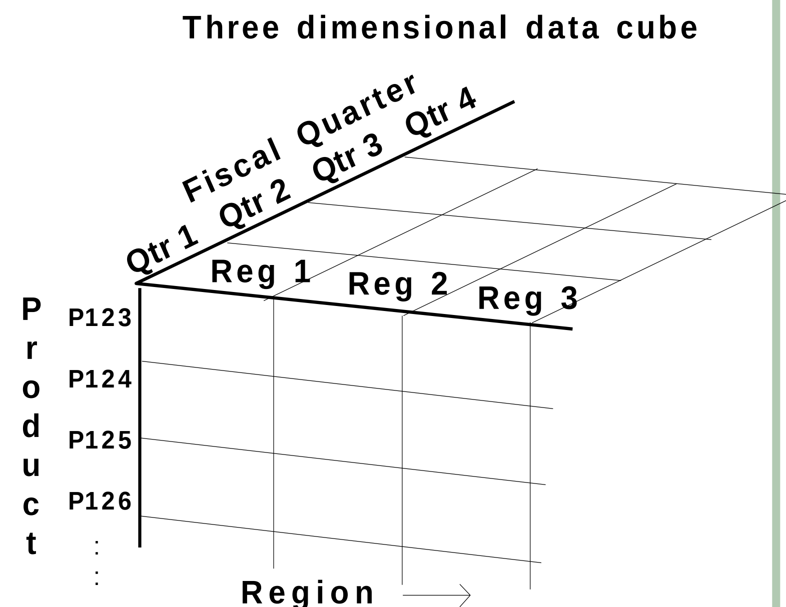
Data Models in Datawarehouse

- Denormalized Data
- Summarized Data
- Multi-Dimensional Data
  - In Data Cubes
  - Instead of data tables
- Details are removed
  - Those data not important for high-level reports

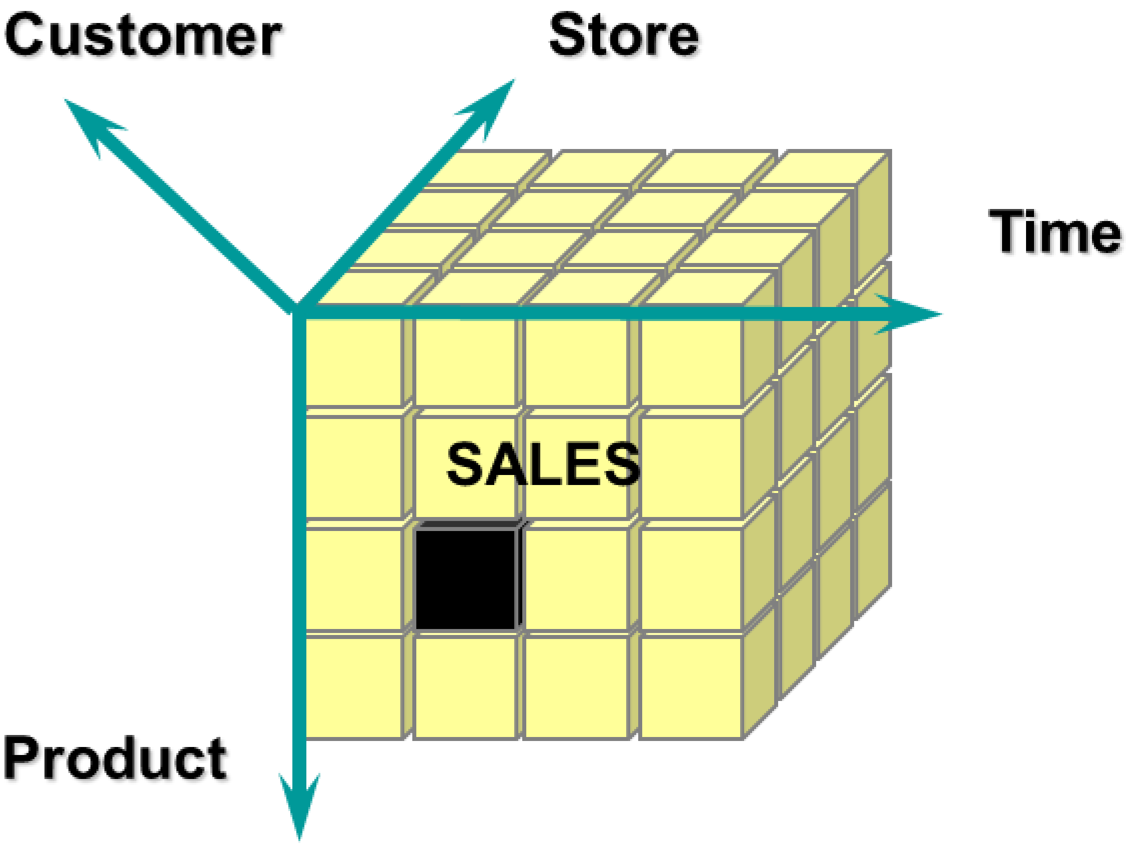
Data Modeling for Data Warehouses

» Example of Two- Dimensional vs. Multi-Dimensional





Data Cubes



Multi-dimensional Schemas

- Multi-dimensional schemas are specified using:
  - **Dimension table**
    - It consists of tuples of attributes of the dimension.
  - **Fact table**
    - Each tuple is a recorded fact.
    - This fact contains some measured or observed variable (s)
    - identifies the measure with pointers to dimension tables.
    - The fact table contains the data, and the dimensions to identify each tuple in the data.

Implementation of Datawarehouse

- Some **DBMS** vendors support datawarehousing and OLAP
  - E.g., Oracle and MS SQL Server
- Many datawarehouse technologies are built upon relational databases
  - E.g., Pentaho
- Some datawarehouse technologies are built on **NoSQL** databases
  - Suitable for big data

Exercise @ Class

- Work with Pivot Tables in Excel

- Pivot.xlsx

Summary: Datawarehousing

- Purpose of Data Warehousing
- Definitions, and Terminology
- Comparison with Traditional Databases
- Characteristics of data Warehouses
- Multi-dimensional Schemas
- Functionality of a Data Warehouse

NoSQL Databases

Motivation

- Relational DBs Cannot Handle **Big Data**
- Relational DBs are good for **structured data**
  - With **predefined** structure
  - And **rare changes** in the schema

- NoSQL:
  - An attempt at using non-relational solutions

The NoSQL Movement

- NoSQL = **Not Only SQL**
  - It is not No SQL
  - Not only relational would have been better
- Use the right tools (DBs) for the job

Origins of NoSQL DBs

- Large scale web-based businesses
  - Google, Facebook, Amazon
- Open source technologies
- Java-based technologies

Definition
from nosql-databases.org

- Next Generation Databases mostly addressing some of the points: being **non-relational, distributed, open-source** and **horizontal scalable**. The original intention has been **modern web-scale databases**. The movement began early 2009 and is growing rapidly. Often more characteristics apply as: **schema-free, easy replication support, simple API, eventually consistent /BASE** (not ACID), a **huge data amount**, and more.

ACID and CAP

- Relational databases support **ACID** transactions
  - **A**tomic
  - **C**onsistent
  - **I**solated
  - **D**urable
- NoSQL DBs relax the conditions by **CAP** theorem
  - CAP: if you want **consistency**, **availability**, and **partition tolerance**, you have to settle for two out of three

NoSQL Values

- ***Basic Availability***
  - The database appears to work most of the time
- ***Soft-state***
  - Stores don't have to be write-consistent, nor do different replicas have to be mutually consistent all the time
  - The information will expire unless it is refreshed.
- ***Eventual consistency***
  - Stores exhibit consistency at some later point (e.g., lazily at read time).

BASE

- An alternative to ACID is BASE:
  - **B**asic **A**vailability
  - **S**oft-state
  - **E**ventual consistency
- Rather than requiring consistency after every transaction, it is enough for the database to eventually be in a consistent state.
- Not all use-cases need ACID transactions

Advantages of NoSQL

- Cheap, easy to implement
- Data are replicated and can be partitioned
- Easy to distribute
- Don't require a schema
- Can scale up and down
- Quickly process large amounts of data
- Relax the data consistency requirement (CAP)
- Can handle web-scale data, whereas Relational DBs cannot

Disadvantages of NoSQL

- New and sometimes buggy
- Data is generally duplicated, potential for inconsistency
- No standardized schema
- No standard format for queries
- No standard language
- Difficult to impose complicated structures
- Depend on the application layer to enforce data integrity
- No guarantee of support
- Too many options, which one, or ones to pick

NoSQL Options

- Key-Value Stores
- Column Stores
- Document Stores
- Graph Stores
- …

Key-Value Stores

- Similar to a Hashmap
- Put(key,value)
- value = Get(key)
- Examples
  - **Redis** – in memory store
  - **Memcached**

Column Stores

- Not all entries are relevant each time
  - Column families
- Examples
  - **Cassandra**
  - **HBase** (Hadoop ecosystem)
  - Amazon SimpleDB

Document Stores

- Key-document stores
  - However the document can be seen as a value so you can consider this is a super-set of key-value
- Big difference: in document stores one can **query also on the document**,

- Examples
  - MongoDB
  - CouchDB

Graph Stores

- Use a graph structure
  - Labeled, directed, attributed multi-graph
    - Label for each edge
    - Directed edges
    - Multiple attributes per node
    - Multiple edges between nodes
  - Relational DBs can model graphs, but an edge requires a join which is expensive
- Example: Neo4j

NoSQL: Summary

- The limitations of RDBMSs
- Motivation for NoSQL
- Definition
- Applications of NoSQL
- CAP theorem


THE
END